

In the Claims:

The following amended claims replace all prior versions of the claims in the application.

1. (cancelled) ~~A method in a network whereby client computers automatically communicate with a server computer or computers such that the client computers automatically perform actions to install, update, deactivate or uninstall software bundles at specific times without any human interaction based on a model created, maintained and modified by a system administrator.~~

2. (currently amended) A method in a network whereby client computers automatically communicate with a server computer or computers such that the client computers automatically perform actions to install, update, deactivate or uninstall software bundles at specific times without any human interaction based on a model created, maintained and modified by a system administrator, ~~A method as claimed in claim 1~~ wherein intelligent client software autonomously works to reach or maintain a state of software bundles specified by the model, comprising:

means for self healing by maintaining the model state at a specified interval by checking and verifying the presence and integrity of every single file and folder specified in the said model;

means of detecting model changes on the server by periodically contacting the said server whereby said intelligent client software automatically updates the said model state to the said model state specified by said server; means of fault tolerant downloading of all data files from said servers whereby the interruptions in data transfers are handled transparently by the said client software such that the said client software will resume data transfer when the network integrity is restored;

means of autonomously downloading and interpreting data files from the said server computers consisting of the user manifest describing the software bundles applicable to the said client, the fileset container describing the contents of the said software bundles and the actual files that form the said software bundle; means of

autonomously activating files in the said software bundle such that the said software bundle is said autonomously downloaded; means of detecting different types of files such as executables, shell scripts or custom installers such that execution of said files is performed by the said client software if endorsed by the said model;

means of autonomously detecting changes in said user manifest after said model changes are committed whereby said client computer software calculates differences of said user manifest compared to said model state of software bundles maintained by said client computer software wherein a said software bundle added difference creates a said download and a said activation wherein a said software bundle remove difference creates a said deletion of the software bundle wherein a said software bundle modified difference creates an updated time schedule for download action, deactivation action or deletion action whereby client computer software distributes processing power from said server computer whereby said server computer is only responsible for replying to download software requests;

means of detecting changes in fileset containers after detection of said model changes whereby said client computer autonomously calculates differences in said fileset containers whereby said client computer software creates said download, said activation or said deletion actions to perform based on said differences in said fileset container wherein said download and said activation actions are performed when said difference reveals a new file difference and wherein a said delete action is performed when said difference reveals a deleted file difference from said fileset container wherein a modification action is performed when said difference reveals a change difference whereby client computer software performs said processing power distribution;

means of reporting all said actions of downloading, activating, deactivating, deleting or updating said software bundle to said server whereby a system administrator creates a status report revealing the success of said state of software bundle action;

means of remotely monitoring the progress of said state of said client software by said system administrator.

3. (original) A method as claimed in claim 2 wherein the server computer creates and maintains fileset containers wherein the server calculates a new fileset container at each committed change of the server model wherein the fileset contains the record identifier, name, size, checksum, permissions and attributes of each file in a software bundle.

4. (original) A method as claimed in claim 2 wherein the server computer creates and maintains user manifests wherein the server calculates a user manifest for every client computer containing a list of all software bundles associated to each client computer as well as the times when the said software bundles should be downloaded, activated, deactivated or deleted by the client computer software.

5. (original) A method as claimed in claim 4 wherein the server computers maintain a model version wherein the model version increases by one after each committed change of the model by the system administrator.

6. (currently amended) A method as claimed in claim 2 [[1]] wherein graphical user interface administrator software displays and modifies the said model, comprising: means for displaying, creating and modifying user accounts; means for displaying, creating and modifying user groups of said user accounts; means for adding or removing said user accounts to or from multiple said user groups; means for displaying, creating and modifying said software bundles wherein modifying a software bundle allows said system administrator to add, delete, move, rename said files wherein said system administrator can modify permissions, operating system flags and said self healing behavior of said files; means for associating said software bundles to said user accounts or said user groups; means of remotely monitoring said server software; means of remotely monitoring said client software; means of remotely modifying said client software settings; means of remotely installing or updating said client software; means of modifying said client software settings on multiple said client computers in a single operation from said system administrator.

7. (original) A method as claimed in claim 6 wherein a said graphical user interface administrator software creates new said software bundles wherein said administrator software creates a pre installation snapshot of said administrator's hard disk and compares the said pre installation snapshot whereby the said system administrator decides on a file by file basis whether said file should be added to the said software bundle.

8. (currently amended) A method as claimed in claim 2 [[1]] whereby said software downloads are distributed across a network with a booster server computer such that said files are autonomously downloaded from said server computer to said booster computer upon request by said client computer software whereby said client computer software requests said file from multiple said booster computers thereby providing fail safe load sharing the said file download operations.

9. (currently amended) A method as claimed in claim 2 [[1]] whereby said software components are designed to run on any operating system that supports TCP/IP network protocols whereby each said client, said server, said booster and said administrator software communicates with said components running on any said operating system.